



Argument against claim rejections under 35 USC 101, statutory subject matter and tangible results are expressed below using any words I wanted. The exact language in the claims is written to fit the 2004 specification and patent rules.

- Optimized machine and network performance through improved definition of performance requirements.
- A patent to evaluate and publish patents.
- Dynamically reconfigurable data structures to perform tasks, solve problems, capture beauty, tell stories, derive and present simpler versions of complex situations.
- Specialized, interchangeable components designed for a logic based language.
- Use of multidimensional waveforms similar to light and sound waves instead of electrical pulses to enable continuity and limit broadcasting range.
- Efficient data processing in underlying structures and systems made by others.
- Freedom from obsolete media and machines while including legacy systems and non-digital materials.
- A way to collaboratively decide which historical materials should be transferred into digital form; maps showing where digital and non-digital materials are stored; definitive guidelines for physical, performance, and environmental requirements to preserve and read media that only works with older machines; ultimately to get rid of the garbage floating around the internet to focus on the best ideas and information in any form.
- Messages with defined conceptual and geographic scope issued to compile precise, relevant, time dependent instructions as intended by originators.
- Tools to address purpose and meaning explicitly.
- Mapping ideas and information to and from multiple places at the same time using streamlined records that reach back to their points of origin.
- A way to avoid unnecessary duplication of computer processing or human thinking about known elements to focus time and efforts on new lines of inquiry.
- Working with network topologies directly to pre-process data, structures and patterns in general ways before actual data and full patterns are retrieved.
- Bridge building techniques to broadly establish new ideas and information.

- A method of reasoning to enable results the invention is designed to support.
- Persistent memory allocation for original pieces of data and their reiterations.
- Standardization procedures to encourage and make creativity, participation and democratic voice easier to accomplish and trace across cultures and disciplines.
- Consistent, expressive documentation, exchange, and tracking.
- A set of rules to follow to play around with, justify, and explore alignments.
- An interface for internationally maintained open source registries of public information versus ways to control the flow to guide the interpretation of private information.
- A technology neutral solution to continue working with partially completed work by others.
- A way of naming, describing, capturing, showing, making, recording, broadcasting, compiling, archiving, and reconsidering that is not typed.

Claims 1 to 10 (Canceled)

Claim 11

“an assembly process for connecting, breaking apart, and organizing data into groups”
as depicted in Figures 5 and 6 using advanced networks of machines to generate
patterned exchange spaces based in logic;

wherein machines calibrate, simplify, and transform patterned groups of ideas and
information into light and sound;

wherein logic and the definition of groups are essential features of system performance.

*Refer to Figure 6, Paragraphs 2.2, 3.4, 3.37, 4.11, Section 7 Use of the Automatic
Evolving Audio and Visual Language and Display Patterns particularly 7.11, 7.36, 7.38,
7.49, 8.2, and Section 9 Data Curation and Digital Preservation particularly 9.12, 9.16,
10.1.*

implementing “a topological record keeping function using virtual forms to representing
relative placement of data components within larger configurations” portrayed in Figures
2, 3, 4, 5, 6, and 9 where topology and new pattern generation and recognition techniques
are applied to digital information context to place historical knowledge and new ideas
into abstract streams as indicated in Figure 3 to see data become entangled, separated
from background information, recognized from different points of view, interrelated and
influenced over time;

wherein relative placement and flow are essential features of system performance.

Refer to Paragraph 1.1, Section 10 Specific Embodiments and Applications.

a shared memory area for registering, intertwining, tracing, and comparing data histories in infinitely large groups organized by context and location;" as outlined in Figures 1, 2, 6, 7, 8 and 10C;

wherein knowledge and display patterns twist, rotate, and become aligned in a stateless space of higher dimensions than those experienced in daily life;

wherein machine and network operations are accomplished using mathematical techniques, modeling products and devices to create, capture, and record the topology of idea and information exchange over time;

wherein shapes and relationships shown by machines are studied through logic, algebra, geometry, knot theory topology and analysis techniques displayed through an evolving automatic language of alignment, density, color, texture and intensity;

wherein the evolving automatic language is an essential feature of system performance;

Refer to Paragraphs 1.23, 2.2, 3.17 through 3.18, 3.33, 4.2, 4.11, 5.3, 7.7, 7.26, 7.28 through 7.29, 7.36, 7.42 through 7.43, 7.47, 8.3, 8.7, 8.11, 9.8

"control mechanisms for overlaying and streamlining similar data and data arrangements until there are no redundancies to enable the identification and ranking of originals;" as illustrated in Figures 3, 5, 6, 7B, and 10C;

wherein topologies merge by eliminating duplicative components with the express purpose of streamlining down to only one high quality original for each component and each topological arrangement, thus changing dynamic shared data stores into sets of high quality maps to originals with no duplicative components or structures;

wherein this step prevents background information from being processed again to display the results of searches, knowledge assembly and investigation tasks that have already been completed correctly;

whereby the only information allowed to persist in dynamic shared data is of high quality, and captured maps between symbols are isolated for review and discussion independently from constant flows of background ideas and information, thus enabling the identification and ranking of originals to help determine which data and data arrangements should be preserved and distributed over the next 1000 years;

wherein the identification and ranking of originals by performing redundancy studies is an essential feature of system performance;

Refer to Paragraphs 1.1, 1.5 through 1.6, 1.8, 1.11, 1.15, 1.18, 1.23, 3.2, 3.4, 3.7, 3.10 through 3.12, 3.15, 3.22, 3.26, 3.29 through 3.30, 3.33, 4.10, 4.18, 5.5 through 5.7, 6.2, 6.7 through 6.9, 7.2 through 7.3, 7.5, 7.9, 7.12, 7.18, 7.20, 7.22, 7.30 through 7.32, 7.35, 7.37, 7.40 through 7.41, 7.44, 7.47, 7.49, 8.3 through 8.4, 8.7, 8.11, 8.17 through 8.18, 8.20, 8.22, 8.26, 9.1, 9.3 through 9.4, 9.6, 9.9, 9.11, 9.13, 9.15, 9.18, 9.21, 10.1, 10.6 through 10.7, 10.13

“an address prioritizing system” executed by representing data in convenient units, then reducing the number of components displayed and processed at a given time by limiting the space patterned ideas and information are perceived to be in;

wherein similar records with overlapping descriptions are examined in a machine and across networks to compare previous user placements, proximities, and priority addresses captured, maintained, and shown within symbols;

wherein symbols are fixed into topologies as characterized in Figure 9 and transformed into multidimensional waveforms for compiling and broadcasting as

set forth in Figure 10;

wherein relative flows of background information versus independent paces of change are revealed over time using links between addresses that are common to more than one symbol to automatically place optimized data and data arrangements on top and in front, making them clearer for broadcasting and compiling;

wherein topologies encapsulate, consolidate and automatically update only the relevant program functions required to read user specified groups of data and data arrangements in self-referring relationships, stamped in context, and made into machine readable copies for future user placement and priority addressing;

wherein address locations that appear in more than one topology are streamlined, united, and made more persistent to be automatically assigned priority being broadcast over networks, wherein this process is managed by concentrating on the display patterns;

wherein persistence is an essential feature of system performance;

Refer to Paragraphs 1.5, 1.13, 1.16, 1.18, 1.24 through 1.25, 2.2 through 2.3, 3.17 through 3.18, 3.20, 3.26, 3.28, 3.30, 4.10, 4.12, 5.2, 6.2 through 6.3, all of Section 6 Monitoring, Controlling, and Influencing Information Placement and Proximity over Time particularly 6.8, all of Section 7 particularly 7.26, 7.32, 7.35 and 7.42, 8.6, 8.17 through 8.21, 8.24, 9.1, 9.4, 9.8, 9.11 through 9.12

“compiler/broadcasters seeking and distributing specific ideas and information by type as recognized by its topology,” as introduced in Figures 10, 1, and 2 when continuous flows of ideas and information are pushed into streams as shown in Figure 3;

wherein Context Driven Topologies are prepared for interpretation by others

when the reference arcs used to structure topologies stretch out, expand, and transform into continuous multidimensional waveforms for compact archiving, distribution, and comparison in purer form;

wherein compacted topologies unfold to become continuous cyclical irregular series of waves;

wherein each arc stays connected to the next arc in a series by changing orientation from the end of one arc to the beginning of the next in the multidimensional waveform state because this is when topologies are purest for machines to rearrange, streamline and compile ideas and information to generate physical patterns ranging from very simple to intricately detailed and incredibly complex;

wherein, unless users are working with the topologies directly, underlying structures and operations are transparent while machines decode, encode, convert and calibrate waveforms and relationships between waveforms based on their topological properties using mathematical patterns, imagery and sounds for recognition;

wherein by default, groups of reference arcs become arranged into spiraling seashell patterns where tighter arcs representing tightly connected components and ideas are pushed to the top, and looser connections automatically fall into the background, the user controls the amount of information presented at one time, in some cases tight connections are too obvious and users prefer to go deeper to make new connections, when this happens even if the components are not changed, the invention is used to reprioritize, restructure and flip the shape to show a topology from a different angle and orientation; these actions are accomplished working through the topologies directly rather than through the original information itself;

wherein default patterns and working with topologies directly are essential features of

system performance;

wherein related portions of multidimensional waveforms are overlapped and combined by machines before the original information is retrieved, if only tight connections are acceptable, only the specified radius and range are recognized, data relationships are also specified and recognized by rotational alignment; therefore, when topologies become interrelated and refer to each other, the radii and rotational alignments begin to streamline and standardize to automate data arrangements for repetitive, specific uses;

wherein knowledge and display patterns to compare and contrast flows of changes over time leads to deep insights and predictions users or machines may not have been aware of until using the invention, thus the right to make the first set of patterns is exclusively claimed to ensure implementation is accomplished as specified;

wherein comparison, contrast, measurement and simplification are essential features of system performance;

wherein, Context Driven Topologies are used to access and interface with machine and network detected errors, establish pathways to growing collections of interesting useful details, simplifying deep patterns in background information not previously observed;

Refer to Paragraphs 1.2, 1.8 through 1.10, 1.15 through 1.16, 1.19, 1.23 through 1.25, 2.2 through 2.3, 2.9, 3.2 through 3.3, 3.5, 3.7 through 3.11, 3.13 through 3.14, 4.1 through, 4.2, 4.6, 4.12 through 4.13, 4.18, 6.6, 6.8, all of Section 7 particularly Paragraphs 7.3, 7.22 7.26, 7.34, 7.40, 7.45, 8.2, 8.3, 8.7, 8.24, 9.1 through 9.7, 9.15, 9.17, 9.20 through 9.21, 10.6, 10.12

“programming standards” as sketched in Figures 2, 3, 4, 5, 6 where reference arcs are running from user propositions and display patterns on machines, transferring in and out

of mathematical frameworks in the stateless space, traveling back to knowledge patterns in published records, and programming advanced networks of machines to delete irrelevant, outdated, misleading, incorrect, and redundant information without a person or research group expending resources to review the same results every time they are presented the same way;

wherein thresholds and tolerances for relevancy, correctness, and aesthetic preference vary by user, research group, machine and network;

wherein documentation of relevancy, correctness, and aesthetic preference is an essential feature of system performance;

wherein thresholds and tolerances are indicated by indelible marks recorded in the symbols to control the flow, persistence, and distribution of multidimensional waveforms;

wherein the symbols, marks, flow, persistence, and allocations of space in shared memory are subject to intense scrutiny, thus enabling greater trust for users to allow machines to decide what the most efficient methods are to consolidate irrelevant, outdated, misleading or incorrect, redundant information before data and data structures are dynamically retrieved or displayed;

wherein programming standards become typical across fields, for example, within the limitations of these claims the American Institute of Architects (AIA) could use the invention to create standard communication patterns for drawings and specifications, capture design intent and aesthetic preferences, and publish up to date building codes in local jurisdictions to control the presentation and retrieval of shared models, ideas and information architects use; wherein the example extends to more groups of users initiating patterns they are responsible for and maintain over time, able to change and update in small increments while still persisting as overall patterns; wherein the example is further extended when the

methods of architects captured in the knowledge and display patterns of architects are applied by other fields to begin their own template, and architects borrow templates from engineers and city planners to incorporate into their communications patterns, whereby programming standards are made stronger and clearer by being allocated more space in shared memory;

wherein using templates and programming standards to allocate space in shared memory is an essential feature of system performance;

wherein data structures are preconfigured using standardized forms to limit the number of data components and forms of display by making components and combinations work together in approved contexts, or prevent data and structures from working or being presented at all;

wherein defining acceptable versus unacceptable contexts is an essential feature of system performance;

Refer to Paragraphs 1.7 through 1.9, 1.11, 1.15 through 1.16, 1.19, 1.21, 2.3 through 2.5, 2.7 through 2.8, 3.1, 3.4 through 3.6, 3.10, 3.24, 3.26 through 3.27, 3.29, 3.33, 3.35, 4.3, 4.6, 4.12, 4.14, 4.18, all of Section 6 Monitoring, Controlling, and Influencing Information Placement and Proximity over Time particularly 6.6, 7.1, 7.10, 7.12, 7.14, 7.16 through 7.18, 7.24 through 7.26, 7.28, 7.31, 7.33, 7.35 through 7.36, 7.40 through 7.41, 7.43 through 7.44, 7.46, 7.49, 8.3 through 8.6, 8.8 through 8.12, 8.19 through 8.20, 8.22, 8.26, 9.1 through 9.3, 9.5, 9.8, 9.11, 9.15, 9.19, 9.21, 10.5 through 10.6, 10.7, 10.12

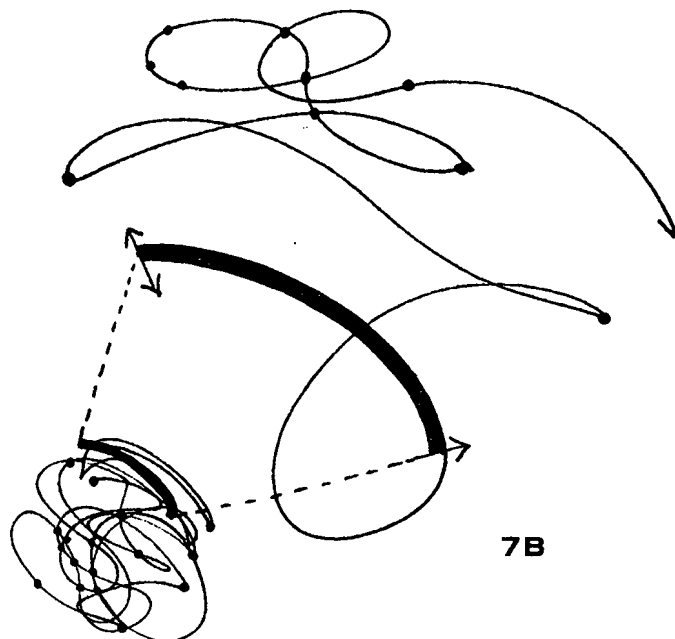
Claim 17.

“wherein boundaries representing information wholes in context are cleaned of potential ideas and information” as indicated in Figures 2, 4, 5, 6, 8, 10A and 10C by opposing knowledge and display patterns in the shared memory area of the stateless space because display patterns are opposite, known twists to knowledge patterns and the two are used in combination to remove ideas and information that do not fit current arrangements;

wherein using knowledge and display patterns in combination is an essential feature of system performance;

wherein Context Driven Topologies represent continuous boundaries composed of arcs that only link to specified components in approved sequences;

wherein existing boundaries are scaled to fit in new user defined configurations as exhibited in Figure 7B;



wherein topologies do not save whole dictionaries of languages, only sets of words in context, nor is space allocated for every processing technique available

in a product or operating system because the only content, techniques, and arcs retained after compression and consolidation into a topological record are limited to only the portions required to precisely regenerate sets of numbers, symbols, words, images, sounds, and underlying structures in specific orders as originators have them placed;

wherein compressing, compacting, allocating space to reflect original placement are essential features of system performance;

wherein users tell machines where to pay more attention to content and techniques known to be correct, preferred, or meeting specified performance requirements;

wherein these ideas and information are assigned more space, resolution, depth and clarity through an interactive data curation process;

wherein users specify the components and measurements to be automatically aligned, proportional, stacked or displayed in known preferred orders;

wherein underlying structures are never automatically aligned or placed without users participation because this is where adjustments occur to make each configuration unique; wherein configurations that are not unique are not duplicated or allocated space;

wherein adjustments and assessment of uniqueness are essential features of system performance;

wherefore as retrieved and placed information receives more attention from users by being continuously displayed, more space and higher resolution are allocated in shared memory by networks and machines;

wherein Context Driven Topologies are built to last, able to be viewed from

different angles, taken apart and put back together until the ideas and information they represent fall out of fashion, are proved to be incorrect or are abandoned causing the data and structures to be compressed into the background and gradually forgotten, rarely accessed, occasionally retrieved and updated to work within current programming standards when users and machines have the knowledge and instructions to seek and extract the essence of original data and data structures by cutting through layers of descriptions which would not be possible without advanced networks of machines role in topological record keeping;

wherein compressed records and instructions are essential features of system performance;

wherein patterns and the automatic language are displayed on machines and in recreated environments where known patterns lead users and machines to components and configurations designed to gravitate, snap, and lock into particular times or logical locations by fixing topologies fixed into a groups, fused into cohesive wholes by machines by being force fit into a single boundaries established by users;

wherein known patterns, gravitating, snapping, and locking into particular times and logical locations are essential features of system performance;

Refer to Paragraphs 2.3 through 2.4, 2.8, 3.7, 3.10, 3.18, 3.21 through 3.22, 3.28, 3.33, 4.10, Section 6 especially 6.7, 7.5 through 7.6, 7.16, 7.20 through 7.21, 7.26 through 7.27, 7.30 through 7.32, 7.42, 7.49, 8.2 through 8.5, 8.13, 8.24, 9.2 through 9.5, 9.11, 9.21, 10.2, 10.4, 10.7

“wherein boundaries are compressed into fixed symbols”

wherein the symbols are a boundary with selectively controlled edges established

in Figures 9 and 10 as if the symbols were characters, but topologies in the symbol state are not exactly like traditional data characters, they are of deeper dimensions and varying densities with related background information stacked underneath;

wherein selectively controlling edges is an essential features of system performance;

wherein fixed symbols are changed back and forth into fluid topologies, where by default, the last way an assembly was displayed before compression into a topology is always the first way it is regenerated; where if subsequent users want to examine ideas and information more closely and carefully, add or subtract components, the topological structure is dissolved and the components become individual objects again so the user can work with the data and structures; wherein subsequent interpretations are made into subsequent topologies which may be closely related to foundation topologies; wherein slight variations are recognized mathematically and perceptually by contrasting past knowledge against current display patterns;

wherein consistency versus slight variations are essential features of system performance;

Refer to Paragraphs 1. through 1.2, 2.4, 2.7, 3.33 through 3.34, 3.9, 3.16 through 3.20, 4.1, 4.18, 7.31, 8.3, 8.14, 9.1 through 9.3, 9.6, 9.11, 9.13

“wherein topologies in the symbol state with massive amounts of information inside yet a small description outside look ready to burst, while symbols with infinite complicated and overlapping descriptions but simple information inside is are wrinkled, yet from far away both look the same;” because boundaries are understood differently between users and machines;

wherein each topology has a strict inside/outside boundary with a patterned space of information flowing around it, machines work on the outside compiling the

mathematical boundaries and users understand the ideas and information captured inside; wherein the system generates evolving records to measure boundaries changing as comprehension evolves;

wherein defining the inside versus outside of boundaries is an essential feature of system performance;

wherein during the time users are manipulating, controlling and selecting context to form topologies, the patterned space around components is not perceived by machines as users perceive - in hierarchies with foreground and background - machines always process all of the data in current arrangements in parallel all the way back to each components point of origin;

wherein perception is an essential feature of system performance;

wherein the term parallel is claimed to literally mean only appearing to converge in the distance from users point of view while not actually meeting in machine records, wherein the exclusive right to implement parallel processing as specified is distinctly claimed to ensure the system performs as intended;

wherein defining points of origin is also an essential feature of system performance;

whereby boundaries are stretched and squeezed to fit in hierarchies created and perceived by users until a conclusion is drawn and context is assigned in the form of a new Context Driven Topology that must be unique or it will not be accepted and given space;

wherein patterns are identified and compared across the entire system by machines regardless of their state of participation (symbol, topology or waveform) in and out of knowledge patterns, shared memory, and display patterns because each Context Driven Topology remains mathematically the same to

machines regardless of the degree of compactness, expansion and space topologies are perceived to be in by users;

wherein machines use mathematical processes and algorithms to order arrangements only as a reflection of users hierarchical structures but machines only process groups and pathways in and out from the stateless space, not hierarchies depicted in the knowledge and display patterns made for users;

wherein variations in interpretations of data and data arrangements over time generates new forms and boundaries made by advanced networks of machines in addition to user contributions, yet topologies are able to be recognized at any point in time and state of completion because the mathematical relationships never change regardless of the ways data descriptions are stretched and squeezed to fit in new or unique contexts over the years;

wherein recognition during varying stages of completion is an essential feature of system performance;

Refer to Paragraphs 1. through 1.2, 2.4, 2.7, 3.33 through 3.34, 3.9, 3.16 through 3.20, 4.1, 4.18, 7.31, 8.3, 8.14, 9.1 through 9.3, 9.6, 9.11, 9.13

“wherein boundaries are made of vectors able to scale without pixilation;” because the topological structures are based on arcs rather than straight lines, corners, or pixels at any time;

wherein the radius of an arc varies according to the looseness or tightness of the connection between data components and placement in data structures when arcs are drawn by users to define connections to arrange the ideas and information being displayed;

wherein some arcs may have such large radius they appear to be a straight line,

but however slight, there is always a curve, adjacent and distant curves may be so overlapped and tangled they do not look like one continuous boundary but a topology is only permitted to be a continuous boundary, otherwise it would just be a set of components rather than a topology;

wherein the drawings set forth the requirement there are no corners between arcs, what may appear as a corner is actually a precise rotation in the way the arcs meet, aligning and measuring radii and rotation is the fastest way for machines to compare data and data structures in general before full data and structures are actually retrieved;

wherein arcs, radius, and rotation for general comparison are essential features of system performance;

Refer to Paragraphs 3.3 and 9.6

“wherein every boundary is continuous and never breaks down while users are interpreting information of different scales”

wherein components or descriptions are never duplicated in a topology, two or more locations in a structure are simply indicated through maps forming the symbols, the degree to which components are considered to be the same is directed by the user, superimposing two or more topologies that share components across scales does not duplicate matching components either, whether machines are compiling in even parallel across all levels, or users are assigning priorities on specified levels, the topologies where users and machines meet twist and rotate in the stateless space for redundant components to align and cancel each other out both in user displays and machine compilations without breaking down because the boundaries are continuous connections;

wherein canceling out across scales and levels are essential features of system

performance;

Refer to Paragraph 1.4, 3.12, 3.31 and Sections 4, 7, and 9, 10.7

wherein the boundaries have no scale because they are every scale; as presented in Figures 5, 10 and 1;

wherein topologies and patterned spaces around them form knowledge patterns moving in a multidimensional abstract stream that is difficult to understand without mathematics or being transformed into the evolving automatic audio and visual language, spaces and display patterns disclosed;

wherein self-referring similarities and patterns in the stateless space begin to develop between symbols and the interrelated ways users create underlying connective shapes characterized in Figure 10C at the end of the process, back to Figure 1 at the beginning of the process over and over again;

wherein self-referring similarities, cycles, and underlying structures are a continuous, essential feature of system performance;

wherein knowledge and display pattern maintain unique mathematical identities able to transfer over generations, recognized primarily by machines comparing and measuring the ways historical knowledge and new ideas come together as they are broadcast and compiled;

wherein ideas and information coming together are made evident through a forced separation of individual change in contrast to a common background of continuous changes over time;

wherein forced separation of changes are essential features of system performance;

Refer to Paragraph 1.1, 2.7, 3.20, 3.34, 4.18, 7.31, 8.3. 9.1 through 9.3, 9.6, 9.12

wherein the boundaries have no inherent thickness, they are built layer by layer or initially connected arc by arc; the mathematical description framework, symbols, signs, and priority addressing of data and data arrangements;

wherein frameworks, symbols, signs, and priority addressing are essential features of system performance;

wherein topologies appear to compress and expand for machines to compile and process in groups where they do not typically belong or easily fit because they do not share past histories and users build bridges to manipulate data and structures to make them fit;

wherein shared memory bridges and manipulation are subject to disputes, challenges and rejection;

wherein as ideas and information persist, working with both knowledge and display patterns cancels out implied thickness for edges to be averaged and revealed as depicted in Figure 3;

wherein determining fit is an essential feature of system performance;

Refer to Paragraph 1.15, 3.9, 3.18, 3.24, 3.33, 6.9, 7.1 through 7.2, 7.18, 7.25, 7.32, 7.35, 7.37 through 7.39, 8.3, 8.6 through 8.7, 8.10, 9.1, 9.6 through 9.7, 9.22

“wherein fixed boundaries serve as descriptions linking information together” as it is streamlining in and out of the shared memory area of the stateless space;

wherein common backgrounds of topologies are organized across networks by sharing component descriptions, similar placement, knowledge components, algorithms, measurements, and histories of interpretation captured in their properties;

wherein the properties of common backgrounds and histories of interpretation are essential features of system performance;

wherein encapsulating, consolidating and automatically updating the specific program functions required to read selected data requires data and the means to interpret them are required to be united to perform within the system;

Refer to paragraph 1.4 through 1.7, 1.11 through 1.12, 1.17, 1.22, 2.1, 2.3, 2.5, 3.9, 3.11 through 3.12, 3.14 through 3.15, 3.20, 3.25, 3.27, 6.3, 7.9 through 7.11, 7.14 through 7.15, 7.21, 7.24, 7.34, 7.43, 8.2 through 8.3, 8.18, 9.6, 9.12, 10.6

wherein the tools and controls for drawing the boundaries

<are as originally stated in the claim>

is accomplished through computer graphics processing, operator interface, and selective ~~visual~~-display; wherein the causes and effects of changing boundaries and fixing them into symbols is accomplished ~~through data and network processing~~ by compiling and broadcasting fixed symbols in contrast to fluid waveforms, thus context is driving the topology of data structures and known topologies are standardizing for specified data uses.

wherein context driving topology is the central feature of system performance;

Refer to the entire specification.

Adjustment to Claim 30, which was improper in form per 37 CFR 1.75c:

Claim 30

~~The system of claims 11, 17, and 26,~~ A signal claim wherein Context Driven Topologies are continuously broadcast and kept alive by being referenced over networks using continuous patterns in lieu of storage media and electrical pulses;

wherein broadcast ideas and information carry both content and technique required to regenerate digital information in context, wherein searchers are automatically led to improved results in context by driving networked topologies to the historically highest priority addresses, master recordings, original high resolution still and moving imagery, sounds, partially interpreted or raw results;

wherein topologies identify the locations of genuine events, objects and living beings;

wherein the topologies are capturing pure mathematical relationships not yet associated with images, sounds, words, or assigned values;

wherein the system is helping ~~us, as~~ individuals and a global society, to working with, weeding out, and controlling data and structures to create bigger pictures, promoting theory and art making on large scales using the forms disclosed for a creative, collaborative processes and strategic record keeping to satisfy a spectrum of user needs from simple tasks to goals as lofty as attempts to ~~and~~ capture elusive beauty;

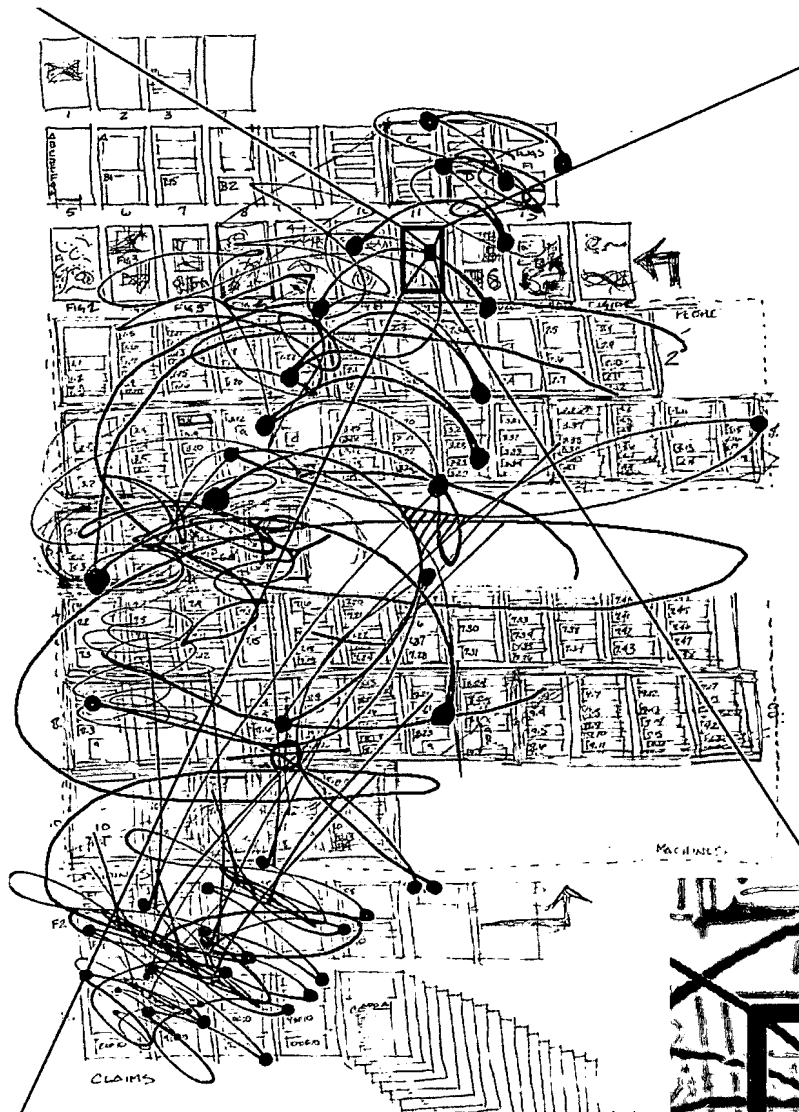
wherefore over enough time, using enough data, the methods set forth show which data and data arrangements are most interesting, correct, unique and worth preserving for further contemplation using new knowledge and new machines in the future.

Refer to paragraphs 3.30, 4.9, 6.7, 7.5, 7.8, 7.10, 7.14, 7.16 through 7.20, all of Section 8 Shared Memory particularly 8.3 through 8.6, 9.4, 10.1

Argument I had possession of the invention, specific locations are provided to comply with 35 USC 112 first paragraph, written description requirement.

Please note, the reason for inserting paragraph numbers in Claims 1 to 10 was to distinctly show locations. Figures 9 and 10 were constructed over the specification to show locations using the only body of text I the author/inventor and you the reader/examiner have in common. In general, Sections 1 to 4 explain requirements for people, Sections 6 to 10 explain requirements for machines. Therefore if machine instructions are what you are interested in, they are likely to be in the second half; the reasoning and ways of working are likely to be in the first half, but the invention is one integrated system.

When we are done with this discussion, I plan make a topology demonstration on my website to show the evolution of my claims, and I want to show the drawings (some which are repeated) inside the claims because these are perfect examples of fluid versus fixed forms of ideas and information.



Claim 11.

an assembly process for connecting, breaking apart, and organizing data into groups;
TEXT LOCATED: 4.11, Section 7 Use of the Automatic Evolving Audio and Visual Language and Display Patterns, and Section 9 Data Curation and Digital Preservation.

drawing reference arcs;

TEXT LOCATED: 3.2, 3.26 through 3.27, 4.9, 7.5, 7.14 through 7.15, 7.40, 7.49,

topological record keeping function using virtual forms

TEXT LOCATED: 1.1, Section 10 Specific Embodiments and Applications

shared memory area for registering, intertwining, tracing, and comparing data histories

TEXT LOCATED: 3.12, 3.21, 4.14, 6.6, 7.49, all of Section 8 Shared Memory, 9.1, 9.3, 9.7, 9.17, 10.13

address prioritizing system

TEXT LOCATED: 1.18, 2.3, 3.17, 3.20, 3.26, 4.12, 5.2, 6.2, all of Section 6 Monitoring, Controlling, and Influencing Information Placement and Proximity over Time, all of Section 7 Use of the Automatic Evolving Audio and Visual Language particularly 7.26 and 7.42, 8.20, 8.24, 9.4, 9.12

Claim 17.

boundaries representing information wholes in context are cleaned of potential ideas and information

TEXT LOCATED: 3.7, 3.10, 3.18, 3.21 through 3.22, 3.28, 3.33, 4.10, Section 6 especially 6.7, 7.5 through 7.6, 7.16, 7.20 through 7.21, 7.26 through 7.27, 7.30 through 7.32, 7.49, 8.2 through 8.5, 8.13, 8.24, 9.2 through 9.5, 9.11, 9.21, 10.2, 10.4, 10.7

boundaries are compressed into fixed symbols

TEXT LOCATED: 1.2, 2.4, 3.33, 3.9, 3.16 through 3.20, 4.1, 7.31, 8.14, 9.11

topologies in the symbol state with massive amounts of information inside yet a small description outside look ready to burst, while symbols with infinite complicated and overlapping descriptions but simple information inside is wrinkled, yet from far away both look the same

TEXT LOCATED: 1.2, 2.4, 3.33, 3.9, 3.16 through 3.20, 4.1, 7.31, 8.14, 9.11

Claim 22. The system of claim 17, wherein mapping back and forth between known and unknown data configurations over time is generating compact portraits of ideas and depicting changes in comprehension;

wherein larger perspectives are enabled for history to stop repeating itself so both people and machines are learning from past mistakes.

TEXT LOCATED: 4.14, 7.29, 7.48, 8.3, 9.1, 9.5, 9.11, 10.7,

Argument against Claim Rejections 35 USC 112 first paragraph, that my invention meets the enablement requirement as pinpointed in the specification below:

Claim 11.

an assembly process for connecting, breaking apart, and organizing data into groups

FIGURES: 5, 6

TEXT LOCATED: 4.11, Section 7 Use of the Automatic Evolving Audio and Visual Language and Display Patterns, and Section 9 Data Curation and Digital Preservation.
drawing reference arcs

topological record keeping function using virtual forms

FIGURES: 2, 3, 4, 5, 6, 9

TEXT LOCATED: 1.1, Section 10 Specific Embodiments and Applications

shared memory area for registering, intertwining, tracing, and comparing data histories

FIGURES: 1, 2, 3, 6, 7, 8, 10C

TEXT LOCATED: 3.12, 3.21, 4.14, 6.6, 7.49, all of Section 8 Shared Memory, 9.1, 9.3, 9.7, 9.17, 10.13

address prioritizing system

FIGURES: 9, 10

TEXT LOCATED: 1.18, 2.3, 3.17, 3.20, 3.26, 4.12, 5.2, 6.2, all of Section 6 Monitoring, Controlling, and Influencing Information Placement and Proximity over Time, all of Section 7 Use of the Automatic Evolving Audio and Visual Language particularly 7.26 and 7.42, 8.20, 8.24, 9.4, 9.12

Claim 17.

wherein boundaries representing information wholes in context are cleaned of potential ideas and information

FIGURES: 2, 4, 5, 6, 8, 10A and 10C

TEXT LOCATED: 3.7, 3.10, 3.18, 3.21 through 3.22, 3.28, 3.33, 4.10, Section 6
especially 6.7, 7.5 through 7.6, 7.16, 7.20 through 7.21, 7.26 through 7.27, 7.30 through
7.32, 7.49, 8.2 through 8.5, 8.13, 8.24, 9.2 through 9.5, 9.11, 9.21, 10.2, 10.4, 10.7

boundaries are compressed into fixed symbols

FIGURES: 9, 10A and 10B

TEXT LOCATED: 1.2, 2.4, 3.33, 3.9, 3.16 through 3.20, 4.1, 7.31, 8.14, 9.11

wherein topologies in the symbol state with massive amounts of information inside yet a
small description outside look ready to burst, while symbols with infinite complicated
and overlapping descriptions but simple information inside is wrinkled, yet from far away
both look the same

FIGURES: 1, 2, 5, 6, 7A, 10A through 10C

TEXT LOCATED: 1.2, 2.4, 3.33, 3.9, 3.16 through 3.20, 4.1, 7.31, 8.14, 9.11

Argument against Claim Rejections 35 USC 112 second paragraph, the invention is defined in the manner required:

Claim 11.

compiler/broadcasters seeking and distributing specific information by type as recognized by its topology;

programming standards for broadcasting specified data configurations in context; wherein, Context Driven Topologies remain mathematically the same and recognizable for parallel machine processing of ~~limitless groups~~ in the stateless space regardless of the ways they are simplified and selectively distributed and displayed in real space and time.

Claim 12. The system of claim 11, wherein representing the pace and extent of changes between different interpretations of the same data are expressed and analyzed through mathematical configurations, or knots of information, in ~~both~~ fixed and fluid forms; as indicated in Figures 1, 3, 5, 6, 7, 9, 10;

wherein fluid ideas and information are made into fixed forms by original authors to be compared and consolidated by machines in shared memory, whereafter subsequent authors working with newer machines make new topologies to feed back into the stateless space as unique fixed forms for subsequent interpretations;

wherein the topologies are an embedded record control system for storing and retrieving information in dynamic relative motion by comparing fixed versions to fluid progress;

wherein the more associations and applications each object accumulates, the more this changes the edges and texture of multidimensional structures and topological boundaries;

wherein the boundaries are inherently dynamic, time and context dependent

forms;

wherein measurable mathematical configurations of knowledge patterns working backwards in time are opposed by display patterns moving forwards in time to expose, streamline and weed out incorrect, temporary, or misleading information from shared data and data structures allocated space in shared memory;

wherein shared data and data structures are essential features of system performance.

Refer to Section 2 Concept Boundaries and the Annotation Process and 3 Symbolic Characters and their Function particularly 3.9 and 3.18, and paragraphs 7.18, 7.32, 7.39, 8.3

Claim 13. The system of claim 11, wherein machines are measuring the ways people are thinking, imagining, and working using the topologies ~~for~~ to tracking which data and data structures have ~~has~~ made sense together in the past;

wherein the system provides machines and networks with arcs to measure, contrasts and comparisons to reflect patterns of human reasoning through radius, rotation, alignment, pace, density, texture, color, intensity and other techniques machines were already capable of measuring accurately in 2004;

wherein the purpose of machines measuring and creating patterns of human reasoning is for machines to help identify and create associations users are not capable of recognizing without machines;

wherein the invention is a new way to show machines related examples and similar versions to explain why some ideas and information are more important than others;

Refer to Paragraph 1.11, 1.23, 2.2, 2.7, 3.6, 3.11, 3.17 through 3.18, 3.33, 4.2, 4.11, 4.14, 5.3, 7.7, 7.26, 7.28 through 7.29, 7.36, 7.38, 7.42, 7.47, 8.3, 8.7, 8.11, 9.8, 10.4

wherein machines and networks are proposing data components and ideal configurations while users are designing and constructing new assemblies;

wherein common histories captured in component descriptions are drawn together because machines always process sets together in parallel rather than hierarchies with foreground and background users need to interact with complex records;

wherein as duplicative components are gradually consolidated, components shift and move to align as machines pull components that share aspects of their histories and descriptions closer together for more efficient processing of similar and repeated units;

wherein machine movement is caused by connections deep in the background unseen by most users except in the ways machine actions affect the display patterns;

wherein unseen connections are an essential feature of system performance.

Refer to Section 2 Concept Boundaries and the Annotation Process and 3 Symbolic Characters and their Function particularly 3.9 and 3.18, and paragraphs 7.18, 7.32, 7.39, 8.3

Claim 14.

wherein introducing new combinations into the shared memory area maintains links between data ~~that may or may not be in proximity.~~

Claim 17.

“wherein the boundaries have no scale because they are every scale” is explained above.

Claim 18.

wherein ~~infinitely large~~ combinations of symbols and component descriptions are overlaid and compared in fixed states;

wherein deriving and assembling histories is compliant with the National Institute of Standards and Technology (NIST) Policy on Traceability contained in the NIST Administrative Manual Subchapter 5.16, online at <http://ts.nist.gov/traceability>; wherein traceability requires the establishment of an unbroken chain of comparisons to stated references;

Claim 19.

wherein consolidating data and network processing techniques ~~typically~~ utilizes the most current version first, except where older versions are required for perceiving data in the method it was made.

Claim 20.

wherein data repeatedly interpreted in the same manner emits simple, strong and clear signals easily identified and found by ~~untrained~~ users looking for information with a specified purpose, or accidentally discovering while wandering around different areas of the shared memory;

Claim 21. ideas we do not understand with ideas and knowledge we do understand

Claim 22. known and unknown

wherein Context Driven Topologies are used to build virtual bridges of understanding to fill in the blanks and bridge the gaps to streamline and compare records of understood ideas and information versus ideas and information that are not understood as depicted in Figure 8 using default spiral forms;

wherein the flow and pace of changes directly corresponds to the pace of changes in knowledge and comprehension idea by idea, relationship by relationship;

wherein users make decisions to place, eliminate, and prioritize data in new data arrangements exemplified in Figures 3 and 4, to use machines and networks to help understand data and data arrangements that are difficult to configure or draw conclusions from by comparing records that are understood against records that are only recently becoming understood and are not well formed yet;

wherein drawing conclusions is an essential feature of system performance.

Refer to Paragraph 1.8 through 1.9, 1.11, 1.13, 2.2, 3.2, 3.11, 3.29 through 3.30, 4.14, all of Section 6 Monitoring, Controlling, and Influencing Information Placement and Proximity over Time, 7.15, 7.42, 8.3, 8.12 through 8.13, 8.19 through 8.20, 8.23, 9.8

wherein larger perspectives are enabled for history to stop repeating itself so both people and machines are learning from past mistakes.

Claim 27.

wherein the language is leading ~~people~~ users and machines to original information precisely regenerated in context.

Claim 24.

wherein concurrent and conflicting interpretations are ~~realistically~~ accommodated gradually causing external descriptions and internal components to influence each others position, eventually canceling each other over time, simplifying the shared memory until only the most accurate and accepted versions persist;

wherein ~~the inventor Deborah L. MacPherson claims~~ the exclusive right to develop the first set of patterns, symbol, and waveform states to run in and out of a stateless space to begin building and further defining the performance requirements for shared memory as

specified herein is explicitly claimed; in contrast, the stateless space can not be maintained, monitored or controlled by any one person or organization, machine or network;

Refer to Paragraphs 1.16, 3.18, 4.7, 7.3, 8.3

Claim 27.

wherein the language is leading people and machines to original information precisely regenerated in context;

wherein Context Driven Topologies evolve as interpretation of data and data structures evolve; the topologies are configured knots of information users understand together over networks, the space around a topology represents the constant flow of ideas and information users do not have the capacity to interpret without machines;

wherein the patterned space around topologies forms and changes over time, where if the patterned space around a knot of information changes, so will the regeneration of the ideas and information in digital form;

wherein the patterned spaces around knots of information are essential features of system performance;

Refer to paragraphs 1.1 through 1.2, 3.11, 5.1, 5.6, 7.3, 7.15, 8.2, 8.9, 9.3, 9.8, 9.15, 10.7

Claim 28. The system of claim 26, wherein Context Driven Topologies making complex digital collections easier ~~for people~~ to

Claim 29. The system of claim 26, wherein Context Driven Topologies are becoming like real objects people form attachments to and begin preferring certain patterns and forms over others;

wherein fine-tuning and control over data and structures that are searched, identified and presented for a particular arrangement of information is directly tied to the users preference and quality assurance needs;

wherein each Context Driven Topology is formed to convey authors ideas clearly by being described and arranged to reflect authors reasoning and meet authors technical specifications and aesthetic preferences;

wherein unique configurations and identities are introduced into the stateless space and automatically gravitates toward certain zones because of threads and connections to related patterns, placements, histories, and configuration types;

wherein the systems mathematical processes are a form of counting and statistics to reflect preferences;

wherein Context Driven Topologies twist, fold, transform, align and associate waveforms, components, structures and symbols that used to only be captured in users imaginations before these relationships are able to be realized through art, science, machines and advanced networks;

wherein Context Driven Topologies are time capsules of ideas and data processing techniques in unique configurations;

wherein human perception, aesthetics, and performance requirements for networks and machines are all affected through optimized performance and improved definitions of performance requirements;

wherein optimization and the definition of performance requirements are essential features of system performance.

Refer to paragraphs 1.1, 1.7, 2.2, 2.6, 3.11 through 3.16, 3.27, 4.6, 4.10, 4.17 through 4.18, 6.9, 7.1 through 7.2, 7.8, 7.23, 7.27, 7.37, 8.1 through 8.11, 8.16 through 8.17, 8.29, 9.1, 9.17, 10.10

Claim 30.

elusive beauty

Modified above.

Argument that 2004/014296, Arno Schaepe et al “Extracting information from input data using a semantic network” bears no resemblance to 10/803040 Context Driven Topologies. Their data structures are picked apart bit by bit to be compared and contrasted versus my invention, side by side, in the order presented. Points of comparison are highlighted different colors.

A system for creating and managing a stateless space for standardizing data and network topologies and integrating computer generated histories wherein data is [REDACTED] [REDACTED] to determine which data persists and continues to be distributed over time.
(See abstract)

A method is disclosed for extracting information from input data comprising mapping of the input data into a data object network. The method uses a semantic cognition network comprised of the data object network, a class object network and a processing object network. The semantic cognition network uses a set of algorithms to process the semantic units. The semantic cognition network defines a processing object in the processing object network by [REDACTED]
[REDACTED]. The processing object comprises the data domain, the class domain and the algorithm. The processing object is used in the processing object network to process the semantic units.

“data object network, a class object network and a processing object network” could not occur in a pure, stateless mathematical space. Defining ideas and information to this degree necessarily places them in context – which is polar opposite to the “stateless space” invented for Context Driven Topologies to stretch out and interact freely over long periods of time. The networks described by Schaepe et al may be obsolete, superseded, and improved in the next ten years, perhaps already obsolete since this is 2004 work. It is unclear how they propose to manage or see mathematical patterns outside their own set ups.

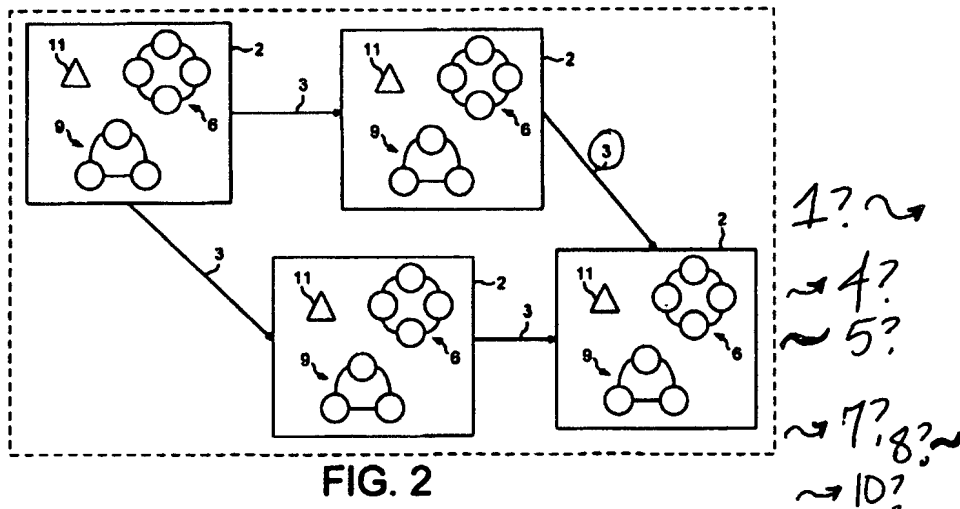
“Integrating” is not the same as “extracting”. Extracting is easy.

“Collectively evaluating” is far more valuable than “selecting a data domain in the data object network, a class domain in the class object network and an algorithm from the set of algorithms.” Each of their selections is a one time, disconnected operation amongst pre-established sets, limited to domains and networks speaking their language and following their rules. The authors do not explain how their system will function over long periods of time in the real world with thousands of standards and regulations, rules and languages. Schaepe et al do not set forth universal methods, or even outline a set of objectives, to reapply and extend knowledge and display patterns to new, concurrent, or barely overlapping ideas and information outside their own purview.

...an assembly process for
connecting, breaking apart, and
organizing data into groups
*(paragraph 24, as process data
into topological context)*

[0024] In the exemplary embodiment described below, a system is described that processes image data. In other embodiments, however, other types of data are processed. For example, other embodiments process data structures provided in a topological context. In addition, yet other embodiments can process data structures in the form of audio data, text data or statistically acquired data (data mining).

“Processing provided data structures” is vague. I could not find the words, images, or intentions Schaepe et al have in mind to “assemble, connect, break apart, and organize data in groups”. If to them “data in groups” means exchanges and recordings within their rigid domains and networks, they still do not specify how their data structures are made or how they are changed by the system. Most importantly, they offer no depiction or further explanation beyond paragraph [0024] about what they mean by topological context.



...marking groups by time, location, and purpose, drawing reference arcs (paragraph 28 and figure 2 arcs)

[0028] FIG. 2 is a schematic representation of a structure of domains and processing objects 2. Each of processing objects 2 comprises a data domain 6, a class domain 9 and an algorithm 11. The various processing objects 2 are connected via linking objects 3 and together constitute a process network. The networking of the processing objects 2 via linking objects 3 defines an execution control. The execution control is a flow control for executing various processing objects 2. The execution control defines which processing object 2 is executed next after the execution of a previous processing object 2 is completed.

Schaepe et al Figure 1 and Figure 2 are elegant in that one operates within the other and every step of their process is not evident in both drawings. I do not consider the screen shots to be drawings. Nevertheless, “connected via linking” shown in the hard edge rectangles designated as 2 all look the same to me. “Marking groups” using Context Driven Topologies is fundamentally different because elements which are exactly the same are not repeated, because this creates unnecessary redundancy. Schaepe et al show an arrangement of 11 (triangle) 6 (four circles connected by one squished implied circle) and 9 (three circles connected by two symmetrical arcs and a straight line). Using Context Driven Topologies, each of these redundant repetitions in a square labeled 2 would be streamlined into one arrangement of four occurrences. If the 11 (triangle) were flipped over, or any one piece or connection was missing in any instance, the discrepancy

would be made evident through algebra, geometry and differential equations, subsequently shown by color intensity, forward or backward placement, crisp or fuzzy edges using data processing techniques described in the specification.

Their links all look the same for deciding and instructing computers and networks exactly which element should be processed next, versus the ones that are already done.

Alternatively “drawing reference arcs” varies in length, radius, and rotation allowing Context Driven Topologies to be reconfigured and reordered into different priorities depending on the task or point of view. I don’t see how this could happen with the inflexible, matching shapes, connections, and linking methods they show because the “execution control defining which processing object 2 is executed next after the execution of a previous processing object 2 is completed” would render the same result every time. How can a computer or network derive which elements deserve priority placement in their Figure 2 when each part and connection is exactly the same? There is no artistic composition, grace, or balance in this drawing.

...a topological record keeping function using virtual forms representing relative placement of data components within larger configurations

(paragraph 10, as mapping data information into a data object)

[0010] In one embodiment, a method extracts information from input data by mapping the input data into a data object network. The input data is represented by semantic units. The method uses a semantic cognition network comprised of the data object network, a class object network and a processing object network. The semantic cognition network uses a set of algorithms to process the semantic units. The semantic cognition network defines a processing object in the processing object network by selecting a data domain in the data object network, a class domain in the class object network and an algorithm from the set of algorithms. The processing object comprises the data domain, the class domain and the algorithm. The processing object is used in the processing object network to process the semantic units.

All of the potential for Schaepe et al “...mapping the input data into a data object network” is destroyed by the typical weak link of a user “selecting” from closed sets. If the options presented are incomplete, out of date, or not the best available for a particular purpose, the other inventors do not explain how advanced networks of machines help to ascertain better fits to assist users in selecting the most appropriate data, objects, and networks for their specified communication or computational goals. Schaepe et al seem to assume all users are fully informed experts with unlimited access to only clean, undisputable facts organized by well established, flawless processing techniques. Context Driven Topologies on the other hand, measure “relative placement”, right and wrong, across a full spectrum of dynamic situations currently underway; placed against backgrounds of similar situations from the past; across entire network topologies using contrast between knowledge and display patterns to gently push users in the right direction. In other words, computers and networks participate in the selection process, which is wide open, interactive, able to be checked and measured at any time. Decisions are not left only in the hands of potentially inexperienced users, providing special utility for data structures understood mainly by machines.

The “topological record keeping function” is implemented by recording sets of established data compositions and pathways in and out of the stateless space. Compositions and pathways proven to be understood by people and computable by machines are perpetuated by being passed around and purposefully brought forward in time. Topological records are formed of trails blazed by experts and worn down through public use. This is different than the other inventors predefined data, networks, and algorithms which, at some point, will run out of energy and become limited in the quality of the answers they are able to generate. Context Driven Topology data structures, on the other hand, are constantly improving, driven by reality, consensus, and logic.

...control mechanisms for overlaying and streamlining similar data and data arrangements until there are no redundancies to enable the identification and ranking of originals
(paragraph 31, as overlay)

[0031] FIG. 3 shows a screen shot of a graphical user interface of data objects 5, class objects 8 and algorithms 11. A user can select data objects 5, class objects 8, algorithms 11, and other settings to implement a desired application. Input data in the form of an image appears in the upper left window of the graphical user interface in FIG. 3. In the same upper left window, the data object network 4 comprising data objects 5 is displayed as an overlay. Individual data objects 5 can be selected and highlighted with an input device. Feature data of a selected data object 5 is displayed in the window labeled "Image Object Information".

“Displayed as an overlay” does not express “control mechanisms - streamlining similar - until there are no redundancies - or - enabling the identification and ranking of originals”. It appears they want to consolidate similar items, but how can machines tell what is similar when everything IS exactly the same? It is unclear in their invention how new data and algorithms are created or introduced into a flow of ideas and information. They do not explain, once an ideal data structure is made using their system, how their version of pathways if they even have a version, fixes structures into measurable topologies. They do not demonstrate how, or if, their structures are able to be precisely regenerated as originally intended using precise, repeatable instructions and common building blocks.

...a shared memory area for registering, intertwining, tracing, and comparing data histories in infinitely large groups **organized by context and location**

(paragraph 23, as extracting information and computer implemented)

[0023] Both a computer-implemented method and a system are disclosed for extracting information from input data. The system can be implemented both on a single computer and on a distributed network of computers such as a local area network (LAN) or a wide area network (WAN). The constituents of a semantic cognition network, also simply called a semantic network, may be implemented in both a **centralized and a decentralized form** of a WAN. As the structure of distributed networks of computers upon which the system is implemented is commonly known in the art, a detailed description of such distributed networks of computers is omitted here.

While “extracting information from input data” serves a useful purpose, especially for microscope and satellite pictures in the areas where Schaepe et al practice. However, it is unclear how a user edits or validates what they are looking at in the information itself. Many unintended, incorrect selections may be made using Schaepe et al’s system because users are only able to select from a limited number of data, networks, and algorithms. A method is not presented for users to redefine problems to seek broader or better options. They make no mention of “registering, intertwining, tracing, and comparing data histories”. Their data and structures are boring and exactly the same. This is not the same as being modular, purposefully designed, and sculptural like art work.

“**Centralized and decentralized form**” seemed promising, however, because items and connections between them exhibit no unique characteristics, what automated viewing and measuring processes can be implemented to “**organize by context and location**”? In Context Driven Topologies, if users seek better data, networks and algorithms inside or outside their own data collections, exchange networks, or kits of algorithms - Context Driven Topologies act like magnets as if the components were high dimensional metal puzzles being put together. If data, networks or algorithms simply do not fit together, they will repel and not stick until the user makes them work by changing and blocking the flow to build virtual bridges, located in 10/803040 specification at (my paragraph

numbers) 1.13, 3.2, 3.29, 7.15, 7.42, 8.3, 8.12, and 8.19. At the same time this is happening, proposed convenient units and typical configurations are constantly being set forth by computers and networks. The highest quality, most agreed upon, pre-computed ideas and information are preserved and brought forward in time for wide spread collaborative adoption, strengthening better data and reliable pathways in and out of the stateless space for future users.

...an address prioritizing system to
characterize data [REDACTED]

in future designs and automated
assemblies

*(paragraph 35, as address and
hierarchical level)*

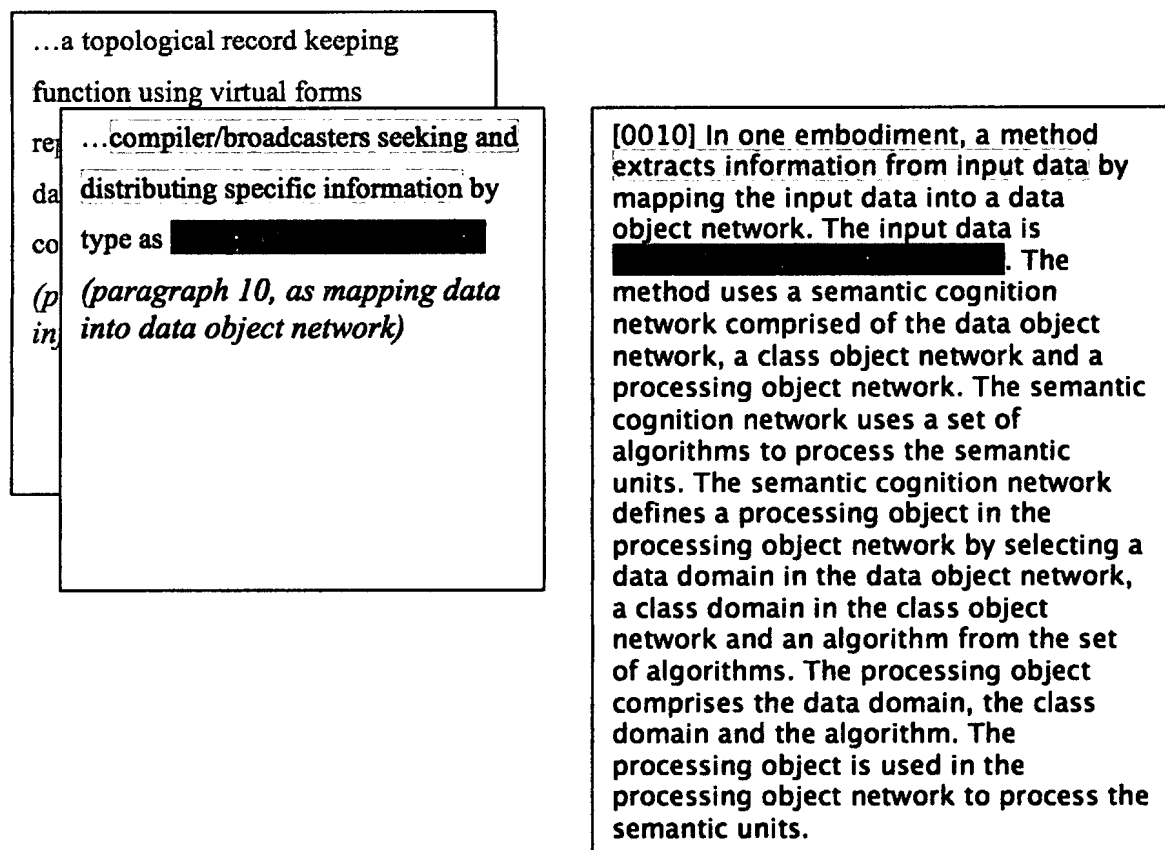
[[0035] FIG. 5 shows a single processing object with a selection list of available data object domains 6. In the dialog element labeled "Domain" there is shown the selection list of available data object domains 6 from which the user can select. The data domain 6 labeled "just execute" executes all linked sub-processing objects, preferably in series. Parallel execution, however, can also be implemented. The data domain 6 labeled "image object level 2" addresses all data objects 5 on a certain hierarchical level of data objects 5. The data domain 6 labeled "neighbor objects (0)" [REDACTED]

[REDACTED]. The data object 6 labeled "sub object (1)" addresses all data objects 5 that are sub-data objects to data object 5, which is being processed by a super-ordinate processing object. The data object 6 labeled "super object (1)" addresses a data object 5 that is a super-ordinate data object to the data object 5, which is being processed by a super-ordinate processing object. As shown in FIG. 5, additional data domains 6 exist and can be defined by the user.

"from which the user can select" does not automate or keep records of an "address prioritizing system"

"labeling" is not "characterizing"

“addressing all data objects 5 neighboring data object 5, which is being processed by a super-ordinate processing object” seems akin to a house, on a street map, in a neighborhood, governed by a global tracker, but missing the critical point of being able to show which data “deserves higher placement and broader distribution”. In other words, if a post office design using concrete blocks works in a Bronx neighborhood, maybe it can also work in an African village using local materials. However, the location to best serve a particular neighborhood is determined by the existing neighborhood, not the global rationale of “post office” or the building methods for individual, repeatable structures.



Looking at paragraph 10 again, “extracting information from input data” is not a “topological record keeping function” as stated above, but it also has nothing to do with “compilers/broadcasters seeking and distributing specific information”. Schaepe et al do not discuss waveforms, varying states of their data structures, or identical mathematical translation between varying modes of interpretation. Nor do they discuss “recognizing by topology” they only talk about “representation by semantic units”. Their invention has to

do with separate parts interacting, mine has to do with wholes changing over time. There are relationships between parts and wholes of course, but Context Driven Topologies are idea and information wholes, greater than the sum of their parts.

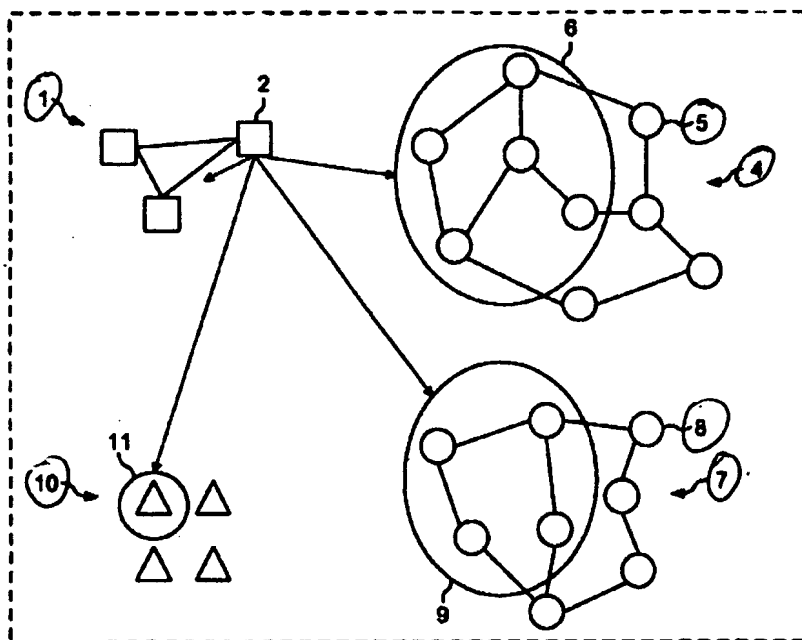


FIG. 1

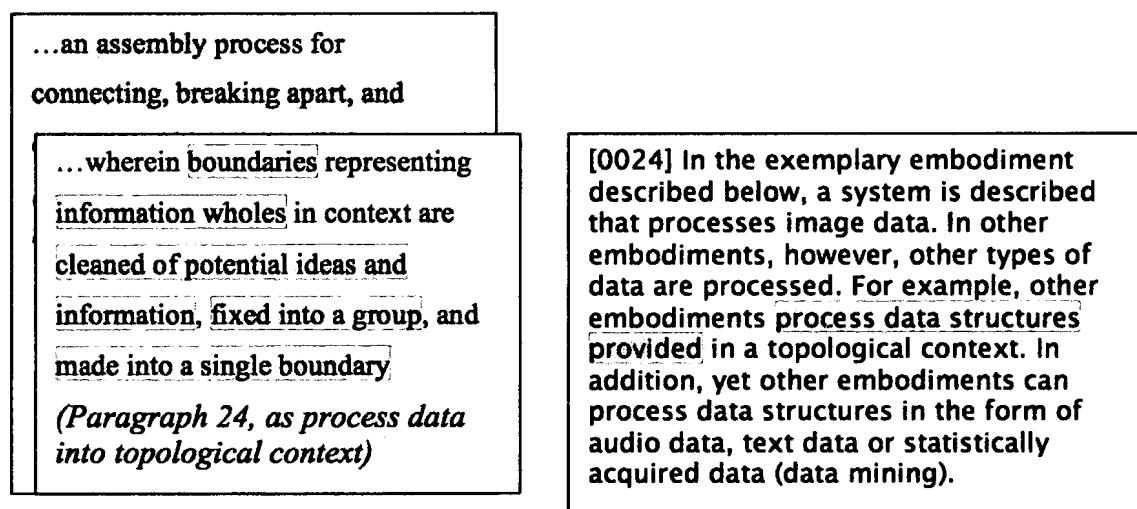
...programming standards for
broadcasting specified data
configurations in context wherein,
[REDACTED]
[REDACTED]
[REDACTED]
[REDACTED] of limitless groups in the
stateless space regardless of the ways
they are simplified and selectively
distributed and displayed in real
space and time
(Figure 1, representation of
semantic cognition network)

[0026] FIG. 1 is a schematic representation of a structure of a semantic cognition network according to an embodiment of the present invention. The semantic cognition network comprises four separate sub-networks: a processing network 1, a data object network 4, a class object network 7 and a set of algorithms 10. All objects [REDACTED] in this semantic cognition network are semantic units.

Figure 1 shows an identification process and kit of parts. However, the “units” of circles, triangles, and squares are lame - made for powerpoint presentations only. “Limitless groups in the stateless space” are beautiful forms composed using texture, color, depth, and proportion.

“**Four separate sub-networks**” is an awkward and clunky representation of, one hopes would be a fluid, semantic cognition network. But their portrayal is sharp and stiff with no organic qualities. One problem is the sub-networks below all look alike so users could become lost, not knowing where they are or what encoded materials they are working with. The overview shown in Figure 1 pulls apart and segregates the squares labeled 2 into their own zone next to data and class object networks and a set of algorithms, the whole system is not integrated, there are **four separate sub-networks**. Alternatively “**specified data configurations in context**” is one system, drawing pure mathematical forms in and out from the stateless space in aesthetic, interactive environments where connections between multiple scales can be seen simultaneously from top to bottom, similar to Photoshop navigator windows and AutoCAD model space.

“**Context Driven Topologies remain mathematically the same and recognizable for parallel machine processing**” means topologies can exist in fluid, fixed, and broadcast forms. In Schaepe et al, if data structures are not already “**contained**” the user cannot generate new ideas, information, and techniques to connect the dots, send out messages seeking potential hole fillers, experiment with the same data in different data structures, or maintain consistent identities across a spectrum of applications and interpretations.



Schaepe et al “**processing of data structures provided**” appear able to be “connected, assembled, and broken apart” moving from the overall system in Fig 1 to the subsystems

in Fig 2, but there is no evidence of continuous “boundaries” across systems or “information wholes”. Most important, they do not talk about how data structures that “almost” work can be “cleaned of potential ideas and information, fixed into a group, and made into a single boundary” able to perform in upper and lower data structures and processing systems.

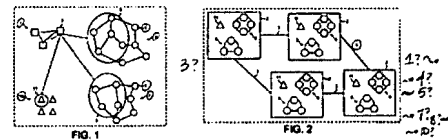


FIG. 1

FIG. 2

...wherein boundaries are decomposed into fixed symbols to serve as a single character in each component existing unique identifiers including existing URLA, museum object numbers, article and publication numbers, geographic locations and place marks, subject matters by area in the shared memory, and adaptable to all information identifiers (Figure 1 and 2)

00261 FIG. 1 is a schematic representation of a structure of a semantic cognition network according to an embodiment of the present invention. The semantic cognition network comprises four separate sub-networks: a processing network 1, a data object network 4, a class object network 7 and a set of algorithms 10. All objects contained in this semantic cognition network are semantic units.

00271 The processing object network 1 comprises a plurality of processing objects 2. The data object network 4 comprises a plurality of data objects 5. A predetermined data domain 6 is chosen from among the plurality of data objects 5, depending upon the given situation as is described below. The class object network 7 comprises a plurality of class objects 8. A predetermined class domain 9 is chosen from among the plurality of class objects 8, depending upon the given situation as is described below. The set of algorithms 10 comprises a plurality of algorithms 11. A predetermined algorithm 11 is chosen from among the set of algorithms 10, depending upon the given situation as is described below.

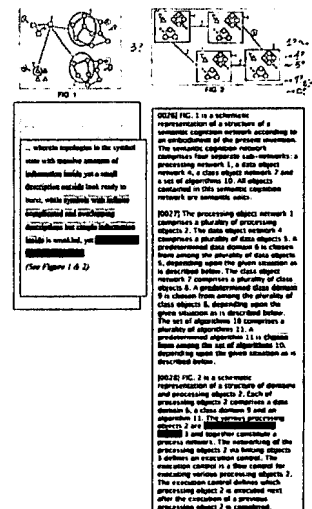
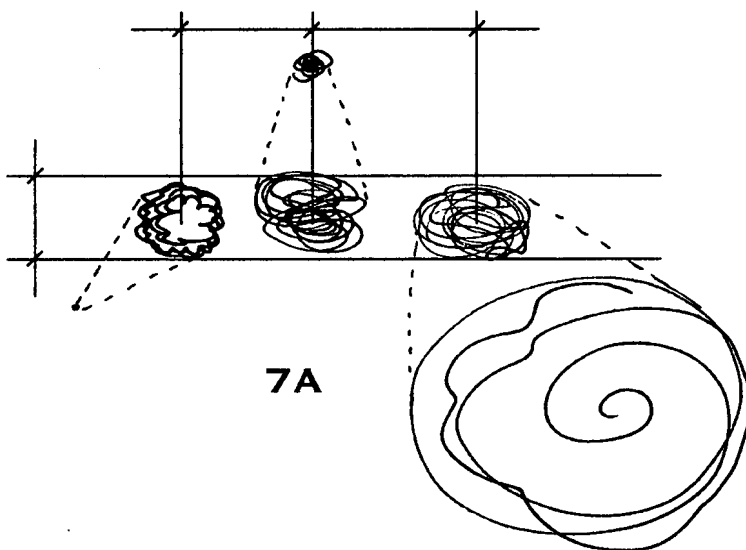
00281 FIG. 2 is a schematic representation of a structure of domains and processing objects 2. Each of processing objects 2 comprises a data domain 6, a class domain 9 and an algorithm 11. The various processing objects 2 are connected via linking objects 3 and together constitute a process network. The networking of the processing objects 2 via linking objects 3 defines an execution control. The execution control is a flow control for executing various processing objects 2. The execution control defines which processing object 2 is executed next after the execution of a previous processing object 2 is completed.

“four separate” does not change fuse into one unlike being “compressed into fixed symbols to serve as a single character”

“adaptable” is not “predetermined”

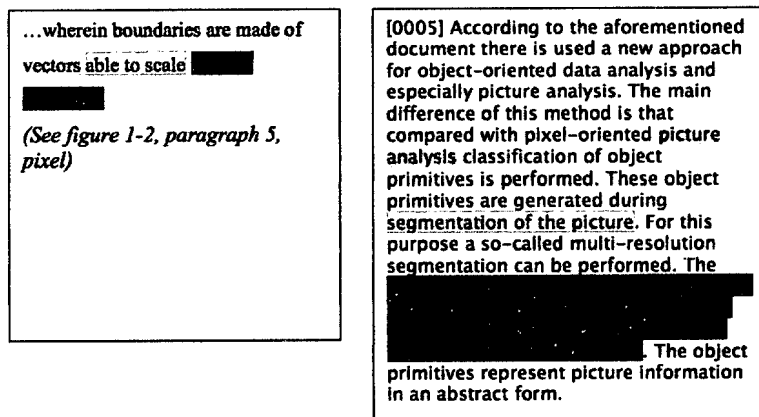
“chosen from among a plurality” leaves out the vital task of deciding what belongs in the pluralities in the first place, which is different than “assigned by information originators and interpreters”

“predetermined class domain” is not adaptable, has nothing to do with symbols or singular units across scales. There is no characterization which would permit topologies in the symbol state to vary so “massive amounts of information inside yet a small description outside” would be obvious. Please refer to Figure 7A instead.



“chosen from among the set of algorithms” once again implies a finite set of solutions, regardless of however large it may be, which offers less to users than infinite possibilities of “symbols with infinite complicated and overlapping descriptions but simple information inside”.

“Connected via linking process” does not permit ideas and information at different scales “from far away to look the same” in Schaepe et al, they are the same.



“Picture analysis” is only one part of “boundaries made of vectors”.

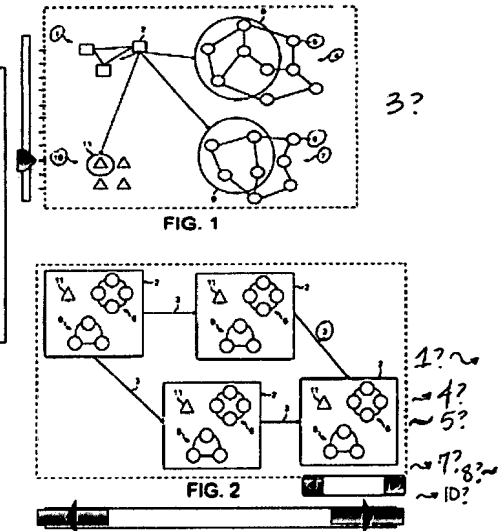
Analysis is after, Context Driven Topology boundaries are also created, this act itself as important as future analysis and perfecting of data structures over time. “Able to scale” is more than “segmentation of the picture”.

“Without pixilation” is made possible by reference arcs forming unique topological constructions. “Multi-resolution segmentation allows for segmentation of a picture in a network of homogenous picture region in each resolution selected by a user” could mean units in an area of interest to be declared a homogenous region, like a political map showing borders of a unified country even if the people there do not all get along or think alike. However, there is no indication how homogenous regions scale.

Slider bars are placed against their drawings to show scaling and measurement of data groups, the structures themselves, and similar topology between groups are all missing. Using Schaepe et al's own subject matters, could users compare microbiology to satellite images to derive new abstract forms or meta-patterns across scales?

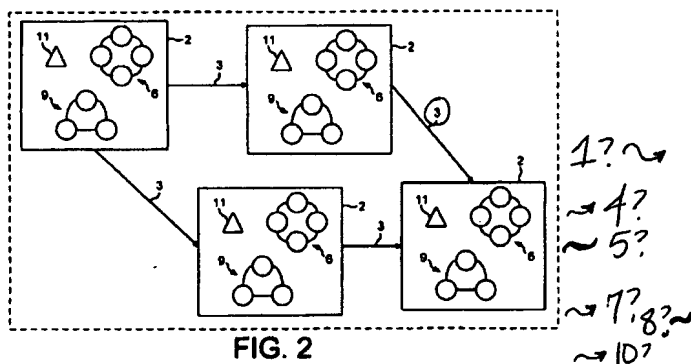
No, "every boundary is continuous" "interpreting information of different scales through the topologies directly" "rather than through the information itself" are not shown their Fig 1 & 2.

...wherein every boundary is continuous and never breaks down while users are interpreting information of different scales through the topologies directly (Figure 1 and Figure 2)



"boundaries that have no scale" are more than "diagrams"

"they are every scale" means able to be represented and processed as deeply connected "domains and processing objects" which Schaepe et al keep separated into systems, sub-systems, classification regions, and existing domains isolated by their own behaviors,



...wherein the boundaries have no scale because they are every scale; wherein the boundaries have no inherent thickness, layer by layer or initially connected arc by arc (Figure 2 arcs)

FIG. 2 is a diagram of a structure of domains and processing objects by the system of FIG. 1.

rules, and languages rather than a holistic unified system based on numbers, measurements and forms any person or machine, or networks of people using networks of machines, can read without ambiguity.

"used" is not the same as "they are built"

“fixed boundaries serving as descriptions” requires more imagination than
 “predetermined data domain 6 chosen from among the plurality of data objects”.

“Streamlining in and out”

of the stateless space
 interlinks data structures
 enabling people to build
 cases, perform their own
 investigations, correct
 history where the facts are
 not straight, promote
 democratic voice, and
 force transparency on
 public information. My
 invention is a longer, more
 continuous, positive
 process than “the
 execution control defining
 which processing object 2 is executed next after the
 execution of a previous processing object 2 is
 completed.”

The “shared memory area of the stateless space”
 continually evolves and improves with new knowledge
 and better machines. Reducing the structural
 dimensions required to discover, participate in,
 capture, record, reinterpret and extend higher and
 higher quality complex work in simpler forms is my
 invention’s primary utility. Schaepe et al are stuck forever with “predetermined
 algorithms 11 chosen from among the set of algorithms 10”

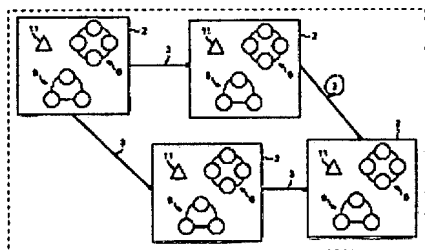


FIG. 2

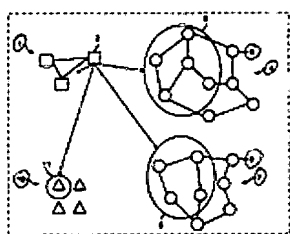


FIG. 1

...wherein fixed boundaries serve as
 descriptions linking information
 together as it is streamlining in and
 out of the [redacted]
 [redacted]
 (Figure 1-2, boundaries)

[0026] FIG. 1 is a schematic representation of a structure of a semantic cognition network according to an embodiment of the present invention. The semantic cognition network comprises four separate sub-networks: a processing network 1, a data object network 4, a class object network 7 and a set of algorithms 10. All objects contained in this semantic cognition network are semantic units.

[0027] The processing object network 1 comprises a plurality of processing objects 2. The data object network 4 comprises a plurality of data objects 5. A predetermined data domain 6 is chosen from among the plurality of data objects 5, depending upon the given situation as is described below. The class object network 7 comprises a plurality of class objects 8. A predetermined class domain 9 is chosen from among the plurality of class objects 8, depending upon the given situation as is described below. The set of algorithms 10 comprises a plurality of algorithms 11.

[redacted]
 depending upon the given situation as is described below.

[0028] FIG. 2 is a schematic representation of a structure of domains and processing objects 2. Each of processing objects 2 comprises a data domain 6, a class domain 9 and an algorithm 11. The various processing objects 2 are connected via linking objects 3 and together constitute a process network. The networking of the processing objects 2 via linking objects 3 defines an execution control. The execution control is a flow control for executing various processing objects 2. The execution control defines which processing object 2 is executed next after the execution of a previous processing object 2 is completed.

“drawing the boundaries” occurs in, and is translated to and from, more structural levels than “data objects.5, class objects 8, algorithms 11”. Even as a developer studio, their boundary configurations are limited to “information relating to the processing object network 1 appears in the window with a tab labeled process”

“**Selective visual display**” is a logical process of scientific elimination and thoughtful design, not simply the portrayal of “**input data in the form of an image**”

...wherein the causes and effects of changing boundaries and fixing them into symbols is accomplished through data and network processing, thus context is driving the topology of data structures and [REDACTED]

(Figure 3, and paragraph 23, as extracting information and networks)

[0016] FIG. 3 is a screen shot of a graphical user interface of data objects, class objects and algorithm objects used by the system of FIG. 1.

[0023] Both a computer-implemented method and a system are disclosed for [REDACTED]. The system can be implemented both on a single computer and on a distributed network of computers such as a local area network (LAN) or a wide area network (WAN). The constituents of a semantic cognition network, also simply called a semantic network, may be implemented in both a centralized and a decentralized form of a WAN. As the structure of distributed networks of computers upon which the system is implemented is commonly known in the art, a detailed description of such distributed networks of computers is omitted here.

“changing the boundaries and fixing them into symbols” is intrinsically different than Schaepe et al’s “data objects, class objects, and algorithm objects”.

Compressing boundaries into symbols fixes and codifies connected series of reference arcs into cohesive, dynamic entities able to cross in and out of related data structures.

“**Extracting information from input data**” is like watching television where the choices are limited and all the programs were all made by other people. “**Standardizing known topologies for specified data uses**” is more like Wikipedia, consensus based standards organizations, knowledge management and good old fashioned craftsmanship.

“Structure of distributed networks of computers upon which the system is implemented
is commonly known in the art” their invention is a repackaging, fine tuning and
advancement of data processing forms and techniques that already exist, on the other
hand “context driving the topology of data structures” is completely new.

<END OF SCHAEPE ET AL ARGUMENT>